1. Demo Scenario Overview

The final demo will demonstrate a functional library catalog.

- Presented features:
    - A homepage that dynamically displays the current time, visit count, and book download count.
    - A book preview page that displays book covers and short descriptions.
    - Editable and interactive comments below the book information
    - A book download page that displays a table that allows users to search and download books in PDF format.
- User actions:
    - Homepage interaction:
        - Check the website visit count
        - Check the book download count
    - Add a comment on the Previews page:
        - Type to insert a short text into the “add comment” field
        - Click the “post” button to submit the comment
        - Can see the previous comments from other users
    - Download a book:
        - Enter keywords to find a book in the search bar → click the “search” button
        - Click the “download” button beside the book entry
        - Confirm the PDF file is served.

- Functional parts:
    - Homepage (index.php) - PHP gets the time and retrieves the visit count and book download count from the database.
    - About page (about.html) - Static HTML/CSS interface that links to other sections.
    - Previews page (previews.php) - PHP retrieves book preview data and renders a gallery.
    - Downloads page (download.php) - PHP loads book records into a searchable table with download buttons.

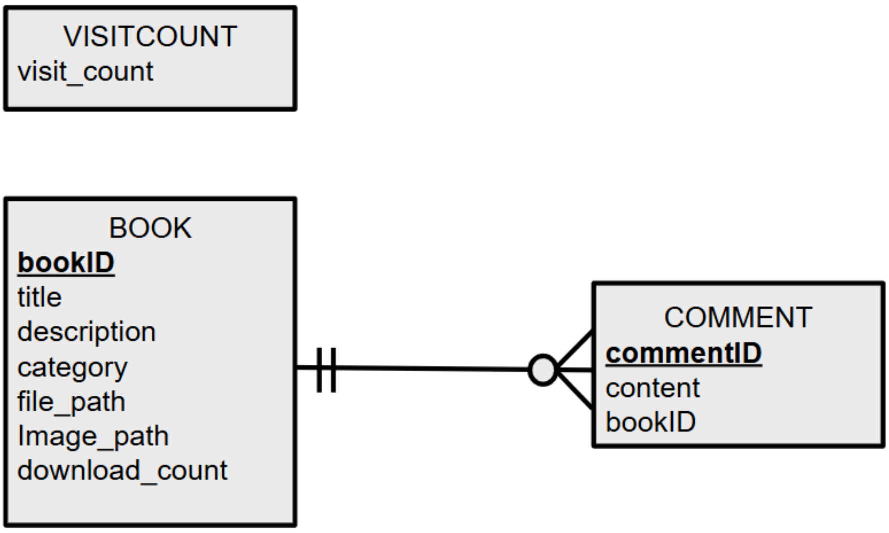
All dynamic data will come from a database, which stores book metadata (titles, descriptions, categories, file paths, image paths), the visit count, and the book download count.

2. Planned URL Endpoints

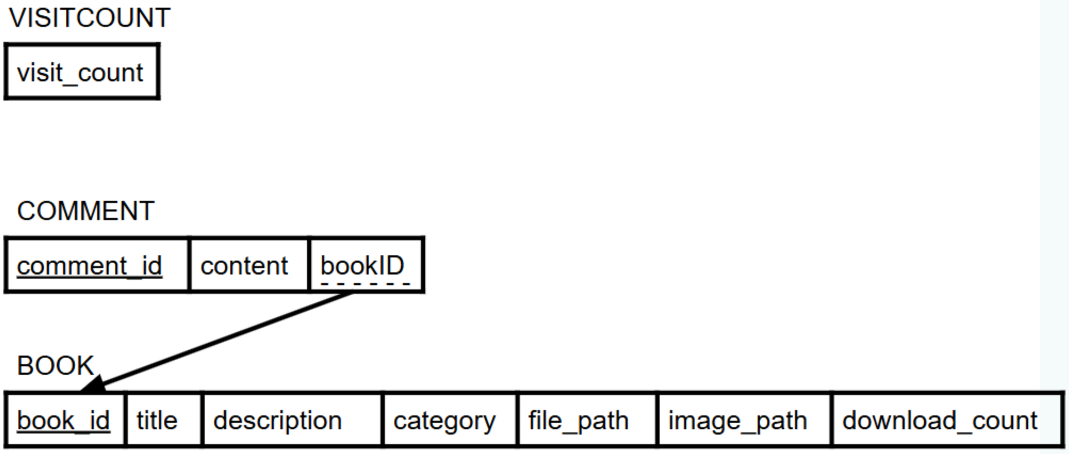
URL Path	HTTP Method	Expected HTTP Variables	Session Variables	Database/Table Operations
/index.php	GET	None	time	● Fetch time (via PHP) ● Read/update visit_count and download_count
/about.html	None	None	None	None
/previews.php	GET, POST	- comment, bookID	None	● Fetch preview data from books ● Insert a new comment into the comment section
/downloads.php	GET, POST	search (via user input), id/filename	None	● Fetch books from a list ● Serve a PDF file based on ID or filename ● Update download count

3. Database Design

A. The Entity-Relationship Diagram (ERD):



B. The Relational Model:



VisitCount table: track the visit count to the website

name	type	note
visit_count	INT	total number of visits to “Home Page” (index.php)

Book table: data of all books

name	type	note
bookID	INT	Primary key
title	VARCHAR(255)	Book title
description	TEXT	Short description of the book

category	VARCHAR(25)	Category
file_path	VARCHAR(255)	PDF files
image_path	VARCHAR(255)	Cover image
download_count	INT	Total number of book downloads

Comment table:

name	**type**	**note**
commentID	INT	Primary key
content	TEXT	What the comment displays
bookID	INT	Foreign key

C. Normalization

The relational model meets the Third Normal Form (3NF) because it has no transitive dependencies and all non-key attributes rely only on their primary key. For example, book information such as title and description relies on the book ID.

A key decision that was made to reduce redundancy was to combine the download count with the book table. This reduces the need to reference a separate table whenever a download occurs.